

Supervision et contrôle des réseaux et services : retour vers le futur

Olivier Festor, Radu State
The MADYNES Research Team
LORIA-INRIA Lorraine

Objectif et contenu

Le domaine de la gestion des réseaux et services s'est développé depuis 1984, date des premières velléités d'aboutir à une approche intégrée, principalement via une action forte de normalisation. Tout au long de ces 15 dernières années de recherche, de développement, de standardisation et de normalisation, le domaine a énormément évolué. De la normalisation initiale qui s'est poursuivie jusqu'en 1995, sont issus un modèle fédérateur, i.e. le modèle Gestionnaire/Agent/Base d'Information de Gestion (MIB) et de nombreux concepts (gestion multi-domaines, services de base, modèles d'alarmes, ...). Depuis, ce modèle a été décliné dans de multiples approches « normatives » (normes OSI, normes de l'UIT-T, standards de l'IETF, documents du DMTF, ...). Pour chacune de ces approches, des solutions logicielles conséquentes (plateformes, toolkits d'agent, ...) pour la mise en œuvre sont ou ont été proposées.

Aujourd'hui, le monde de la gestion des réseaux et services est multi-normes, chacune tentant de couvrir une niche, tout en empiétant le plus possible sur la niche des approches voisines. Si les raisons de cette évolution sont évidentes, le résultat, i.e. une multitude d'approches à la fois concurrentes et complémentaires est pour le moins déconcertant. C'est dans ce contexte de crise de la normalisation, de révolution des fondements du domaine face à l'émergence conjointe de nouvelles technologies support mais également l'expression croissante de nouveaux besoins que se place notre intervention.

L'objectif de l'exposé est triple : (1) donner au participant une vision précise de l'évolution du domaine sur les 15 dernières années, (2) décrire les approches émergentes et les confronter aux besoins des services en forte croissance, (3) présenter les défis que la recherche en supervision et contrôle des réseaux devra relever dans la prochaine décennie.

Pour atteindre cet objectif, la présentation est organisée comme suit. Tout d'abord, nous dressons un bilan des évolutions de la gestion des réseaux et des services de

2 Nouveaux paradigmes et architectures pour la gestion des réseaux et services

1988 à 2002. Ce bilan est assorti d'une cartographie précise de l'état actuel des concepts ainsi que des offres normatives et technologiques du domaine.

Sur la base de cette cartographie et de la présentation des besoins naissant dans les différents domaines d'application : réseaux, intergiciels, services, nous présentons les approches de supervision qui sont aujourd'hui les plus adaptées pour répondre à ces exigences. Nous détaillerons notamment :

- la gestion basée sur la technologie Java avec une illustration au travers de l'approche JMX ainsi que par les interfaces ARM 3.0,
- la gestion utilisant les technologies du W3C et du Web (XML, http, SOAP, ...) et sa mise en œuvre dans la gestion de la configuration (SyncML et JunoScript, et des Web Services (WSLA).
- la gestion à base de politiques.

Si ces approches sont aujourd'hui en plein évolution, il apparaît qu'elle ne peuvent en l'état, couvrir tous les domaines d'activité dans lesquels la supervision est requise. Ceci nous amènera à identifier les défis sous-jacents et à présenter les pistes de recherche poursuivies aujourd'hui à travers le monde ainsi que dans notre équipe.

Guide de lecture & références

Le support du cours (transparents au format PDF) est disponible sur le site du projet MADYNES (<http://www.madynes.org>). En complément du support, nous donnons ci-dessous un guide de lecture permettant à tout participant d'approfondir sa recherche d'informations sur les thèmes abordés dans cet exposé.

Les conférences IFIP/IEEE IM (ISINM jusqu'en 1995, Integrated Management depuis 1997), IEEE/IFIP NOMS (Network Operations and Management Symposium) et le workshop IFIP IEEE DSOM (Distributed Systems : Operations and Management) représentent la source majeure de contributions de très grande qualité de la recherche en gestion de réseaux et services. La collection des actes de ces événements depuis 1989 (1^{er} ISINM) constitue une base solide pour toute recherche dans le domaine.

En complément, les volumes de JNSM (Journal of Network and Systems Management), des workshops comme LISA, IEEE IPOM, IEEE MMNS ainsi que des sessions dédiées ou des numéros spéciaux d'autres événements, journaux ou

magazines en réseaux et services, sont également très intéressants (JSAC, GLOBECOM, INFOCOM, ACM SIGCOM, Communications Magazine, ...).

Les normes ou assimilées étant légion, le lecteur trouvera également une masse d'informations sur les sites des différents organismes et/ou consortiums (UIT, IETF, DMTF, JCP, W3C, Open Group, SyncML, Telemangement Forum, ...).

Finalement, les références ci-dessous ont, en plus des documents cités ci-dessus, largement alimenté le contenu de la présentation et complètent de nombreux points abordés.

- [Festor01] O. Festor, *Ingénierie de la gestion de réseaux et de services : du modèle OSI à la technologie active*, Habilitation à Diriger des Recherches de l'Université Henri-Poincaré – Nancy 1, December 2001.
- [KellerL02] A. Keller, H. Ludwig, *The WSLA Framework : Specifying and Monitoring Service Level Agreements for Web Services*, Proc. LISA'2002, Philadelphia PA, USA, July 2002.
- [Lewis01] L. Lewis, *Managing Businesses and Service Networks*, Kluwer Academic Press, 2001.
- [MartinFlat02] J-P. Martin-Flatin, *Web-based Management of IP Networks and Systems*, John Willey & Sons Publisher, 2002.
- [OpenGroup01] The Open Group, *Application Response Measurement, Issue 3.0 – Java Bindings*, The Open Group, July 2001.
- [SahaiMSMC02] A. Sahai, V. Machiraju, M. Sayal, A.v. Moorsel, F. Casati, *Automated SLA Monitoring for Web Services.*, Proc. 13th. IFIP/IEEE International Workshop on Distributed Systems : Operations and Management, DSOM'2002. M. Feridun, P. Kropf, G. Babin, Eds. LNCS 2506, Springer, Montréal, Canada, October 2002.
- [Schönwälder02] J. Schönwälder, *Evolution of Open Source SNMP Tools*, Proc. SANE 2002 Conference, May 2002.
- [SturmMJ00] R. Sturm, W. Morris, M. Jander, *Foundations of Service Level Management*, Sams Publishing, 2000. (compléments sur www.nextslm.org)
- [FleuryL01] M. Fleury, J. Lindfors & al., *Jmx: Managing J2ee Applications with Java Management Extensions*, Sams Publishing, 2002.

Supervision et contrôle des réseaux et des services

Retour vers le futur !

Olivier Festor, Radu State
The MADYNES Research Team
LORIA – INRIA Lorraine
{Surname.Name}@inria.fr

MADYNES
<http://www.madynes.org>

Plan

- Supervision des réseaux & services
 - Quelques rappels
- Évolution des espèces
 - Du GDMOSAure au DISMANopithèque
 - Leçons !
- Les solutions actuelles & à venir
 - JMX,
 - WBEM & CIM
 - ARM
 - XML Management
- Synthèse
- Futur
- Q&R.

Supervision des réseaux et des services

Définition

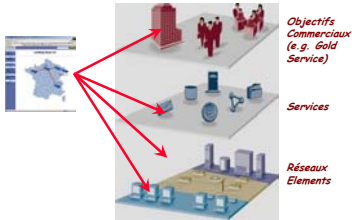
Ensemble des **activités** mises en œuvre pour le **respect des contraintes de qualité** et les coûts des services offerts aux usagers

Besoins

- Surveiller (observer)
- Contrôler (agir)
- Planifier l'évolution

Aires fonctionnelles

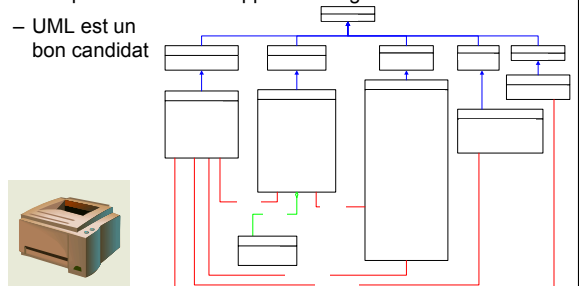
- Fautes
- Configuration
- A..comptabilité
- Performance
- Sécurité



Modèle de données (Data Model)

Quelle information est utile pour la gestion ?

- Modèle abstrait des interfaces de gestion
- Indépendant de toute approche de gestion
- UML est un bon candidat



Modèle de l'information (Information Model)

Modèle de l'information adapté à une approche de gestion

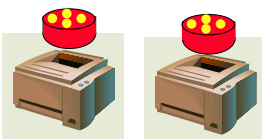
- Langage & modèle de spécification dédié
- Nommage spécifique
- Laison forte avec le protocole

```
hrDeviceStatus OBJECT TYPE
SYNTAX INTEGER {
    unknown(1),
    running(2),
    warning(3),
    testing(4),
    down(5) }

ACCESS read only
STATUS mandatory
DESCRIPTION "The current status of the device ...."
::= { hrDeviceEntry 5 }

hrPrinterStatus OBJECT TYPE
SYNTAX INTEGER {
    other(1),
    idle(3),
    printing(4),
    warmup(5) }

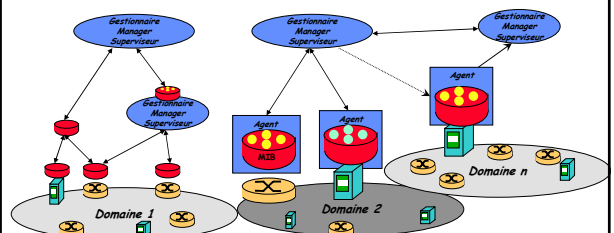
ACCESS read only
STATUS mandatory
DESCRIPTION "The current status of this printer device ...."
::= { hrPrinterEntry 1 }
```



Modèle organisationnel

Quels intervenants sont nécessaires suivant quelle organisation ?

- Superviseur, Agent, hiérarchie, proxy, médiateur, ...
- Domaines physiques, logiques



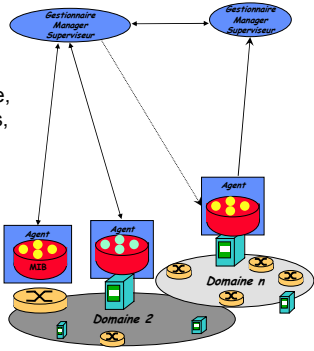
Modèle de communication

Quels services sont mis en œuvre ?

- Lecture d'attribut, affectation, atomicité, requêtes, portée, filtrage, invocation de messages, remontée d'alarmes, ...

Sur quels protocoles ?

- TCP, UDP, IP, 802.3
- ANEP, TP0, ...



Opérations & Applications

Modèle fonctionnel

Base

- Monitoring,
- journalisation d'alarmes,

Avancées

- fourniture de VPN en ligne,
- Reporting
- Accounting,

Plan

Supervision des réseaux & services

- Quelques rappels

Évolution des espèces

- Du GDMOsaure au DISMANopithèque
- Leçons !

Les solutions actuelles & à venir

- JMX,
- WBEM & CIM
- ARM
- XML Management

Synthèse

Futur

Q&R.

Evolution des espèces

Problème initial (1984)

- Seules des solutions propriétaires existantes
- Frein à l'hétérogénéité des équipements réseau

Besoin de l'époque

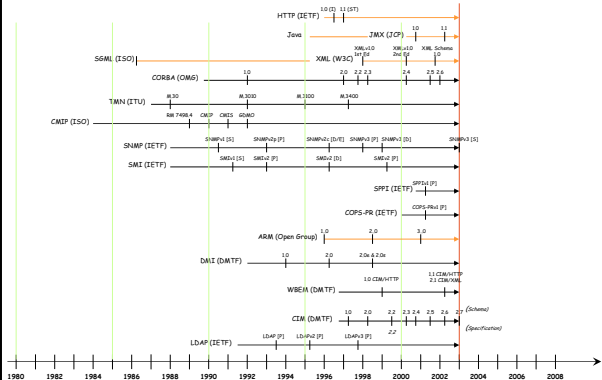
- Disposer d'une approche de supervision intégrée

Solution : Normaliser

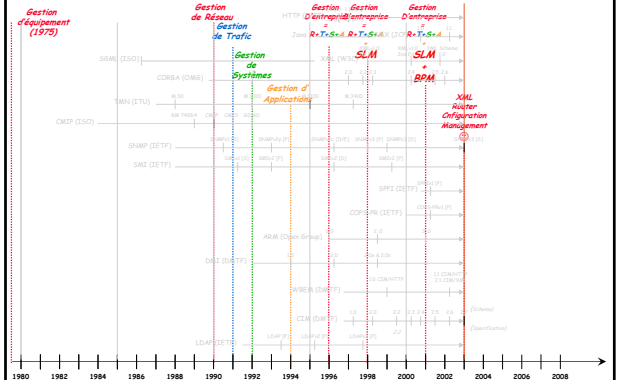
- 2 organismes s'y attellent
 - ISO-OSI, CCITT-UIT-T
- Dès 1990, l'IETF propose SNMP
 - "Une approche **SIMPLE** de transition vers l'approche OSI qui répond aux besoins immédiats"

Depuis tout a dérapé...

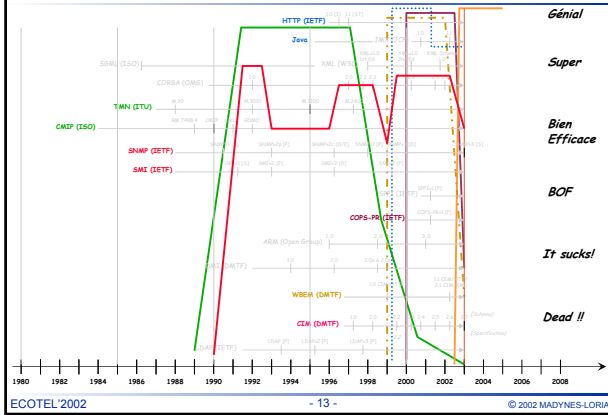
Historique (extension de la grille de [Schönwälder02])



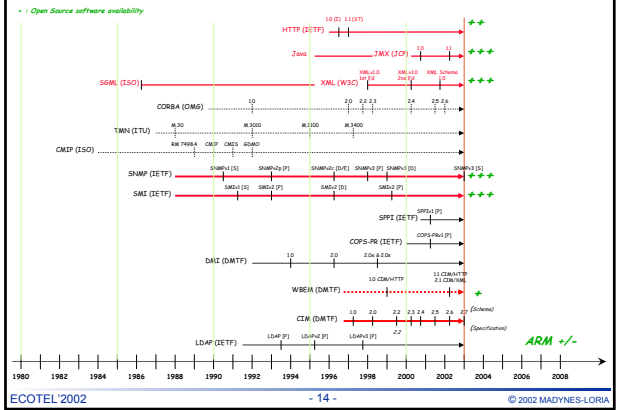
Phases de gestion ([Lewis01])



Hype



OpenSource focus & survivabilité

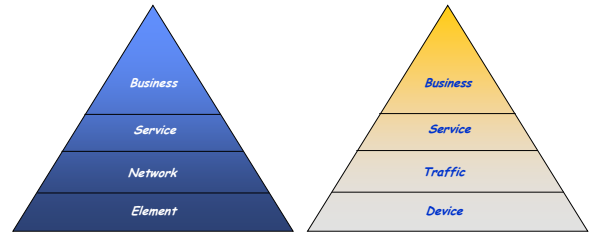


Leçons du passé

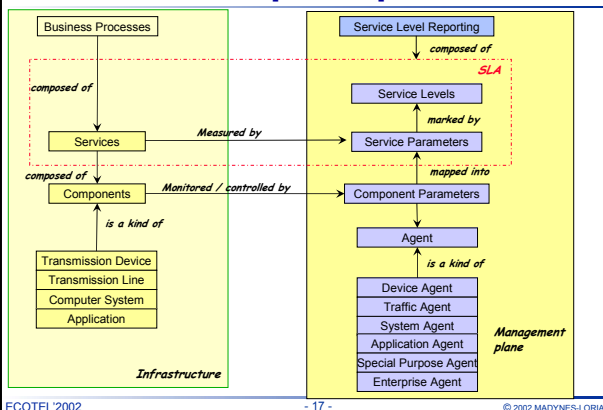
- On ne se débarrasse pas de son passé !
 - Ou du moins pas facilement
- Et encore, nous n'avons pas comptabilisé les cadavres
 - DME, FMA, SNMPConf®, JMAPI, TINA, ...
- La normalisation a été une bonne idée en gestion de réseaux :
 - Oui :
 - elle a donné un cadre général à la gestion
 - Fab4 : Organisation / Information / Communication / Function
 - Fab4 : Device / Network / Service / Business Management
 - Elle a parfaitement atteint son rôle social
 - Elle a permis à nombre d'entre nous de découvrir le monde
 - les réunions ISO à Sydney, ça le fait !
 - Les réunions IETF à Osnabrück, moins ☹
 - Non :
 - Elle a fait perdre un temps pharaonique à des légions d'ingénieurs, de chercheurs et autres espèces binaires.
 - Elle en fait perdre encore (standardisation maintenant)
- Ne survivent « vraiment » que les approches disposant d'une base logicielle Open Source stable ?
 - C'est peut-être l'inverse ...

Fab4 : TMN & Internet

- Manager/Agent/Proxy/Adaptor
- Information Model



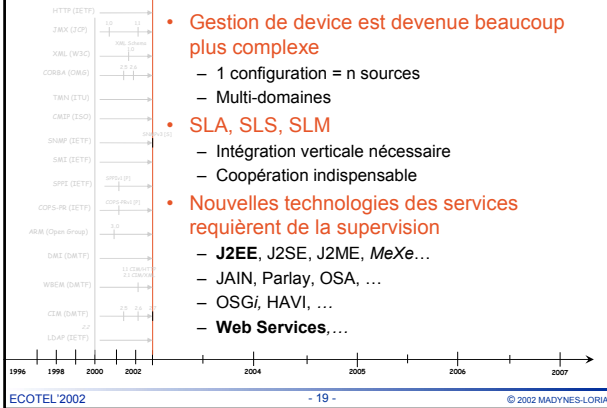
Gestion de services [Lewis01]



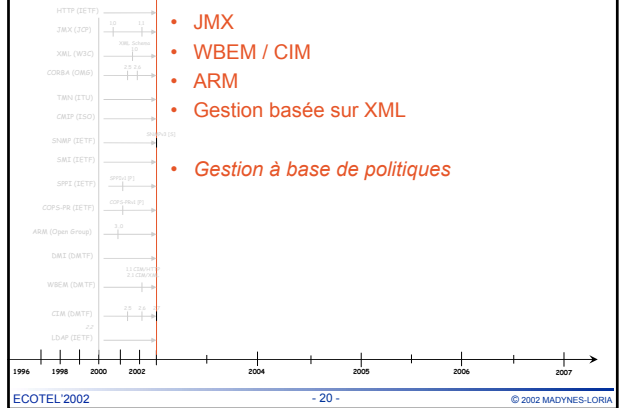
Plan

- Supervision des réseaux & services
 - Quelques rappels
- Évolution des espèces
 - Du GDMOsaure au DISMANopithèque
 - Leçons !
- Les solutions actuelles & à venir
 - JMX,
 - WBEM & CIM
 - ARM
 - XML Management
- Synthèse
- Futur
- Q&R.

Et maintenant ?



Quelles solutions technologiques à court / moyen terme

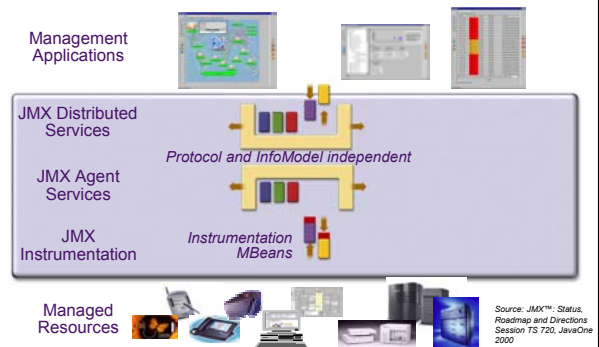


JMX : Java Management Extensions

- **Architecture d'agent de supervision en Java**
 - Permet l'instrumentation d'applications/composants Java en Java
 - Favorise l'utilisation de Java pour de l'instrumentation de composants non-Java
- **Modèle d'Information**
 - Ensemble d'objets MBean
- **Objets gérés**
 - Nom + MBean
- **Nommage : à la X.500**
 - *Mais pas tout à fait quand même !*
- **Découplage instrumentation / export / protocoles d'accès**

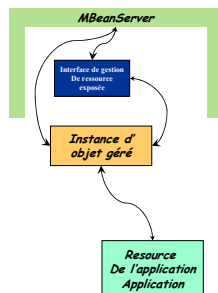


JMX architecture globale



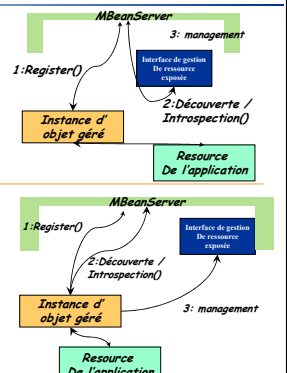
JMX : Objets gérés

- **Vision simplifiée**
 - Une interface de supervision = MBeanInfo
 - Attributs exposés
 - Méthodes « invocables » sur interface de gestion
- **3 niveaux de programmation d'objets**
 - Qui construit l'interface exposée ?
 - L'agent par introspection
 - L'application à la demande de l'agent
 - L'agent par ordre externe
 - Quelle est la souplesse de l'interface ?
 - Définie à la compilation
 - À l'exécution
 - Qui construit l'objet géré support de l'interface de gestion
 - Le programmeur de l'application
 - L'agent délégué par l'application



JMX : Objets gérés

- **MBean standard**
 - Interface de gestion définie à la compilation
 - Découverte par l'agent à l'enregistrement de l'objet géré
 - + / -
 - modèle simple
 - interface figée à la compilation
- **MBean dynamique**
 - Interface de gestion construite par l'objet géré au MBean
 - Découverte déléguée par l'agent à l'objet
 - + / -
 - Plus lourd à implémenter
 - MBeanInfo, get, set, invoke, ...
 - Interface de gestion adaptable possible (le fameux conditional packages de l'OSI ☺)



JMX : Objets gérés

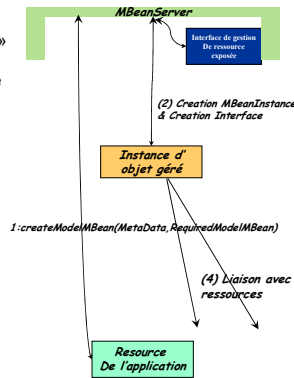
• MBean Model

- Ressource délègue le « codage » du MBean au MBeanServer
- Interface décrite par la ressource lors
- + / -

- Modèle pour faînénants
- Descripteurs & fonctions avancées
 - Gestion de cache
 - Persistance
 - ...

• MBean Ouvert

- MBean Dynamique avec restriction sur types:
 - Paramètres de méthodes
 - Types de retour



JMX : un petit exemple

```
// Interface
public interface PrinterMBean {
    public Integer getMaxCopies();
    public void setMaxCopies(Integer n);
    public PrinterStatus getStatus();
    public void reset();
}

// The MBean
public class StandardPrinter implements PrinterMBean {
    private Integer supCopies = 2;
    private PrinterStatus fStatus;

    public Integer getMaxCopies() {
        return supCopies;
    }

    public void setMaxCopies(Integer pSubCopies) {
        fSupCopies = pSubCopies;
    }

    public PrinterStatus getStatus() {
        return fStatus;
    }

    public reset() {
        (setStatus("down", "other")); sendReboot();
    }

    public Printer() {
        fStatus = new PrinterStatus();
    }

    // not visible at the Mgmt Interface
    public Integer getTemperature() {
        return fTemperature;
    }
}
```

```
public class MonApplication {
    // The agent
    private MBeanServer myMBeanServer =
        MBeanServerFactory.createMBeanServer();

    public MonApplication() {
        CommunicatorServer.registerMBean(htmlAdaptor, null);
        try {
            ObjectInstance htmlAdaptorInstance =
                myMBeanServer.registerMBean(htmlAdaptor, null);
            ObjectName mbeanObjectName = new
                ObjectName("madynes:MonPremierMBean");
            myMBeanServer.createMBean(
                "PrinterMBean",
                PrinterMBean);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

JMX : Services d'agent

• MBeanServer

- Conteneur de MBeans
- Assure le nommage & enregistrement des MBeans

• Services avancés

- Monitoring
 - Attributs d'objets
 - Valeurs numériques et chaînes de caractères
- Timer
 - Périodique, unique
- MLet
 - Chargement dynamique de MBeans
- Query
 - Requêtes de sélection sur MBeans
 - Portée + Filtre à la OSI
- Relationship

JMX : modèle d'événements

• Evénements sont une partie intégrante de l'approche

• Modèle Event/Listener de Java

- Objets émetteurs
 - broadcaster
- Objets consommateurs
 - Souscription

• Limité au sein d'un MBeanServer pour le moment

```
public class StandardPrinter implements PrinterMBean,
    NotificationListener {
    {
        public void handleNotification(Notification notification, Object
            handBack) {
            System.out.println(notification.getMessage());
        }
        .....
        public PrinterMBean(SystemMBean systemMBean) {
            (....)
            NotificationFilterSupport nf = new
                NotificationFilterSupport();
            nf.enableType(new String[]{"PC:alarm"});
            systemMBean.addNotificationListener(this, nf, nf);
        }
    }
}
```

```
public class SystemMBean extends NotificationBroadcasterSupport
    implements MyPCMBean {
    {
        .....
    }
}
```

JMX : composants des événements

• Notification (CL can be subclassed)

- type: chaîne de caractère qui définit le type de la notification ex. fr.loria.pc.alarm
- seq number: dans le contexte de la source
- estampille
- message: chaîne décrivant la cause
- userData: données additionnelles fournies par l'émetteur (Java Object)

• NotificationListener (IF)

- handleNotification(Notification n, Object handback)

• NotificationFilter (IF)

- isNotificationEnabled (invoqué par le broadcaster)

• NotificationBroadcaster (IF)

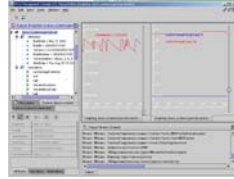
- getNotificationInfo: liste de toutes les notifications que la source peut émettre
- add/RemoveNotificationListener
- sendNotification

JMX et consors dans le JCP



JMX : plateformes

- **Plateformes Open Source**
 - MX4J (mx4j.sourceforge.net)
 - Implémentation complète du toolkit
 - Plusieurs extensions
 - Connecteurs additionnels
 - Support de méta-données (MBeansInfo) dans les MBeans standard
 - Licence Apache
 - Open JMX : prédécesseur de MX4J
 - JBOSSMX :
 - Le cœur de la plateforme J2EE
 - JBoss
 - Un détournement génial du but de JMX
- **Autres**
 - TMX4J
 - Tivoli, alphaworks
 - JMXRI
 - Implémentation de référence
 - JDMK
 - Plateforme Sun



Utilisateurs JMX (source JMXperience)

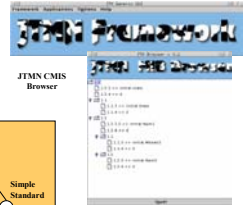
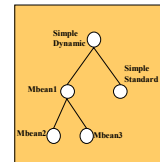
- AdventNet: Agent Toolkit + Java/JMX Edition + Manage Engine + Middleware Manager + Middleware Manager WebLogic Edition + WebNMS
- BEA: WebLogic
- Compiere Open Source ERP & CRM
- Critical Path: Internet File Server, Presentation Server, Registered Mail Server, System Console
- CSC (Scandinavia): LABKA II
- Dirig Software: Dirig Agent
- Hewlett Packard: Core Services Framework + HPAS + OpenView
- IBM/Tivoli
- IBM: Web Services Toolkit 3.1. + WebSphere Business Components + WebSphere Business Integrator + WebSphere Voice Server
- Innovative Systems Design: ITVerity
- IONA Technologies PLC: iPortal + Orbix + Orbix E2A XMLBus Edition Technology 2.0
- iReasoning Networks: iReasoning JMX SNMP Agent BUILDER
- The Jakarta Project (Apache): Phoenix
- JBoss
- Log4j
- Lutris: EAS
- Macromedia: FlashMX, JRun 4
- Manage.com: FrontLine Java Management Edition (JME)
- Media style GmbH
- Misys International Banking Systems Ltd: Meridian
- ObjectWeb: JOnAS 2.5, JORAM 3.1
- Pramati Technologies
- Resonate Inc: Resonate Commander
- Schmid Telecommunication: Pegasus Element Manager
- Sonic Software: SonicXQ
- SpiritSoft: SpiritWave 5.1 + SpiritWave Integration Server
- Sun Microsystems: Java Dynamic Management Kit (JDMK) + Netra CT Managed Object Hierarchy (MOH) + Netra HA Suite + Netra T1 + DReAM, Distributed Resource Allocation Manager + SunONE Application Server + SunONE Portal Server
- Sybase: EAServer 4.0
- TCC: Rexp AppServer 1.0
- Tomcat
- Wily Technologies: Introscope
- Xadra's VelocityAdaptorServer
- Zareus, Inc: Zareus Application Platform

JMX : + & -

- **Un mélange génial**
 - Des Javavores, des OSiristes
- **Résultat:**
 - La gestion OSI sans ses défauts
 - Un toolkit Agent
 - Trivial à l'usage (Learning curve idéale)
 - D'intégration facile (voir exemple précédent)
 - Des implémentations OpenSource de grande qualité...
 - ...surtout MX4J et MC4J, TMX4J, ...
- **Quelques limites ?**
 - Pas de langage ni de modèle de spécification de modèles d'information neutre
 - Si : Java... mais est-ce vraiment une tare ?
 - Les agents sont forcément en Java ... Super ☺
 - Partie cliente (Manager) ou du moins l'accès à distance (remoting) tarde à venir (JSR 160)
 - Approches propriétaires d'accès aux agents pour le moment.

Autres choses faites ou faisable avec JMX

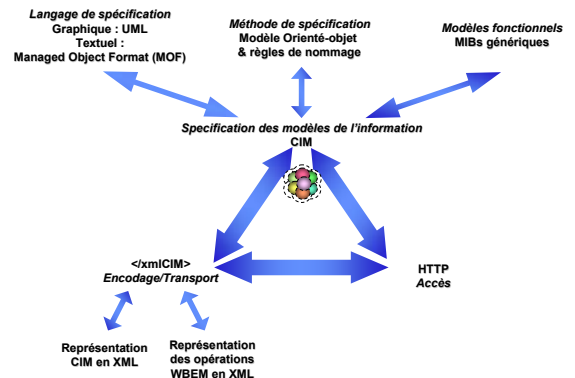
- **Agent JMX comme agent OSI**
 - Passerelle CMIS/JMX



WBEM : l'approche du DMTF

- **Web-Based Enterprise Management**
 - But :
 - approche d'unification de la supervision et du contrôle à l'échelle de l'entreprise ☺
 - Le chapeau ultime.
 - Approche :
 - Modèle neutre, modèle organisationnel, modèle de communication, modèle fonctionnel (modèle neutre)
 - Technologies du Web : XML & Web
 - Approche *traditionnelle* de la normalisation d'un modèle de supervision
- **DMTF : les ambitions**
 - Avant WBEM : Desktop Management Task Force
 - Après WBEM : **Distributed** Management Task Force

WBEM : les composants



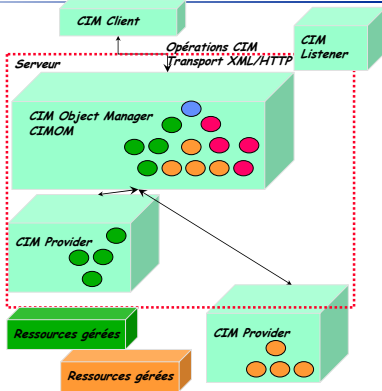
Architecture standard de WBEM

• CIMOM

- Agent d'interface avec clients
- Agent d'agrégation pour les providers
- Intègre les opérations CIM
- Comporte des services avancés
 - Journalisation, persistance, ...

• Providers

- Fournisseurs de données de supervision
- Instrumentent les ressources



CIM : principes

• Modèle orienté-objet

- Héritage simple
- Classes, instances

• Nommage hiérarchique

- Méta-modèle + instances sont dans la MIB

```
// Reference vers une classe
HTTP://dolcourt.loria.fr/root/cimv2:CIM_ComputerSystem
// Reference vers une instance
http://dolcourt.loria.fr/root/cimv2:CIM_ComputerSystem.Name="dolcourt.loria.fr"
```

CIM : un exemple MOF

```
[Version ("2.7.0"),
Description("Capabilities and management of the Printer...")]
class CIM_Printer : CIM_LogicalDevice {

    [Description ("Status information for a Printer,..."),
    ValueMap {"1", "2", "3", "4", "5", "6", "7"},
    Values {"Other", "Unknown", "Idle", "Printing", "Warmup", "Stopped Printing", "Offline"},
    MappingStrings ("MIB.IETFPrinter-MIB.hrPrinterStatus")]
    uint16 PrinterStatus;

    [Description ("An integer array indicating the types of paper supported."),
    ValueMap {"0", ... "75"}, Values {"Unknown", "A2", "A3", "A4", ...}]
    uint16 PaperSizesSupported[];

    [Description ("Sets the power state of the Device.") // Héritée de LogicalDevice
    uint32 SetPowerState(
        [IN, ValueMap {"1", "2", "3", "4", "5", "6"},
        Values {"Full Power", "Power Save - Low Power Mode", "Power Save - Standby",
        "Power Save - Other", "Power Cycle", "Power Off"}]
        uint16 PowerState,
        [IN] datetime Time);
}
```

CIM : service d'accès au CIMOM

• Opérations décrites indépendamment du transport

- Opérations sur MIB (extrinseques)
 - Accès aux classes, instances, attribus
 - Instanciation, destruction d'instance, de classes
 - Accès aux attributs, qualifieurs, ...
 - Exécution de requêtes
 - Primitives d'interrogation des relations

```
void SetProperty ( [IN] <instanceName> InstanceName,
                  [IN] string PropertyName,
                  [IN,OPTIONAL,NULL] <propertyValue> PropertyValue=NULL
                  )
```

- Invocation de méthodes dans des objets (intrinseques)
 - Appel de méthode sur une instance d'objet

• 23 primitives de service

- Qui a dit que CMIS était compliqué ?

CIM & remoting XML/HTTP

• DTD représentation composants CIM en XML

- Attribut, valeur, classe, instance, qualifieur, ...

• DTD de représentation des opérations

- getClass, getInstance, ...

• Intégration dans un transport HTTP

- Choix de l'encapsulation
- Choix des primitives de service
- Extensions éventuelles d'HTTP

WBEM HTTP : un exemple

```
M-POST /cimom HTTP/1.1
HOST: madynes.loria.fr
Content-Type: application/xml; charset="utf-8"
Content-Length: xxxx
Ext.Cache-Control: no-cache
Man: http://www.dmtf.org/cim/mapping/http/v1.0; ns=73
73-CIMOperation: MethodCall
73-CIMMethod: GetProperty
73-CIMObject: root/cimv2
<?xml version="1.0" encoding="utf-8" ?>
<CIM CIMVERSION="2.0" DTDVERSION="2.0">
  <MESSAGE ID="87872" PROTOCOLVERSION="1.0">
    <SIMPLEREQ>
      <IMETHODCALL NAME="GetProperty">
        <LOCALNAMESPACEPATH>
          <NAMESPACE NAME="root">
            <NAMESPACE NAME="myNamespace">
              <LOCALNAMESPACEPATH>
                <PARAMVALUE NAME="InstanceName">
                  <INSTANCENAME CLASSNAME="MyDisk">
                    <KEYBINDING NAME="DeviceID">
                      <KEYVALUE>C:</KEYVALUE>
                    </KEYBINDING>
                  </INSTANCENAME>
                </PARAMVALUE>
                <PARAMVALUE NAME="PropertyName">
                  <VALUE>FreeSpace</VALUE>
                </PARAMVALUE>
              </IMETHODCALL>
            </SIMPLEREQ>
          </MESSAGE>
        </CIM>
```

WBEM : implémentations

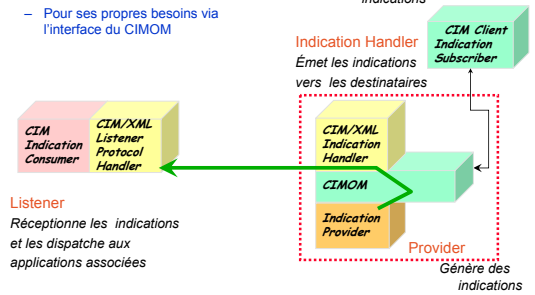
- **OpenPegasus** THE OPEN GROUP
 - C++, multiplateformes (Winxx, Linux, Solaris, ...)
 - Environnement complet
 - CIMOM, Provider
- **Windows WMI**
 - CIMOM Windows NT4 SP 4 & up
- **HPUX CIMServer**
 - Instrumentation HP-UX & Tru64
- **WBEM Services**
 - Java CIMOM (Sun)
 - Sun SCSL License
- **OpenWBEM**
- **SNIA WBEM**
- **Standards Based Linux Instrumentation for Manageability (Sublime)**
 - Linux Grid Management NCSA SB&IM
 - CIM-based distributed Monitoring
 - SNIA CIMOM + Interface native C pour provider
 - Instrumentation Complète du système Linux

WBEM : notifications

3 modes de souscription

- Directement au lancement du CIMOM
- pour le compte d'un tiers via l'interface du CIMOM
- Pour ses propres besoins via l'interface du CIMOM

Client
Souscrit pour le compte d'un tiers ou de lui-même à des indications



WBEM Notifications

- **Souscription**
 - Instanciation d'un handler
 - Qui comporte la destination
 - Instanciation d'un ou plusieurs filtres
 - Sous forme d'une requête
 - Extension avec scope des sources
 - Association du filtre au handler
- **Les objets de la souscription sont eux-même des objets de la MIB**
 - Vous vous souvenez les *Event Forwarding Discriminator* dans la gestion OSI ?
- **Modèle abstrait d'indications**
 - Alertes (modèle OSI like)
 - Traps SNMP (sous-classes de ProcessIndication)
 - Indications CIM (création/destruction de classes/instances, modification de valeurs d'attributs...)
 - Toute indication est sous-classable
- **Transport HTTP**
 - M-Post du handler vers les listeners
 - CIM Listener : Serveur HTTP + Processeur

WBEM : synthèse

- **Une approche de gestion traditionnelle**
 - Un modèle de représentation des informations de ressources
 - Un nommage global
 - Un langage de spécification
 - Un modèle organisationnel
 - Client / CIMOM / Provider / Listener
 - Un protocole : CIM Operations
 - Un transport : HTTP / XML
 - Un modèle fonctionnel
 - CIM Core, Common, ...
- **Un support d'intégration dans l'architecture d'agent**
 - CIMOM
 - Provider
 - [Qualifieurs] sont super utiles pour cette intégration
 - Une implémentation d'intégration SNMPv1 dans WMI
- **C'est une approche très inspirée par le meilleur de la supervision...**

WBEM : + & -

- **+**
 - La gestion OSI avec les outils d'aujourd'hui
 - Tout est extensible grâce aux qualifieurs « maison »
 - Des modèles « béton » (de très grande qualité)
 - C'est le précurseur du Web-based Management non propriétaire
- **-**
 - Interopérabilité difficilement crédible aujourd'hui:
 - Interfaces CIMOM/Providers non définies ...libre à chaque implémentation
 - Transport XML/HTTP a encore qq gros trous
 - MOF extensible.... Bonjour les parsers !
 - #pragma
 - Qualifieurs à gogo

WBEM : évolutions

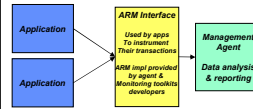
- **Migration du transport vers SOAP**
 - Travaux en lien avec OASIS-Open
- **Support d'un mécanisme transactionnel**
 - Locking, commit, rollback
- **Interface provider/CIMOM formelle**
 - Des propositions existent via les travaux de l'Open Group
- **Sécurité**
- **Remoting ... tient.tient**
 - Découverte de CIMOM & providers,
 - registration, opérations asynchrones entre providers
- **Améliorations**
 - Traitement d'erreurs, nommage d'instances, mécanisme de requêtes, ...
 - (voir présentation de A. Westerinen au DMTF devcon02 <http://www.dmtf.org/>)

Compléments : EMMA

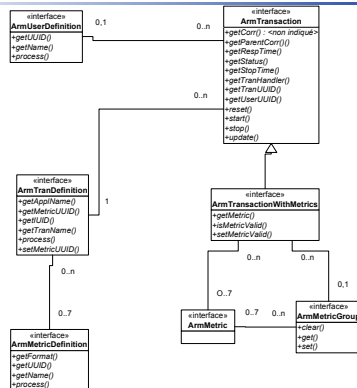
- Présenté à Java One 2002
- Exemple type de complément JMX/WBEM
 - Surcouche sympathique de génération d'agent de gestion Java
 - Indépendante des frameworks
 - Génère du JMX (Agent + MBean) &/v WBEM (Provider + managed objects + spécification MOF)

ARM : simplicité, efficacité

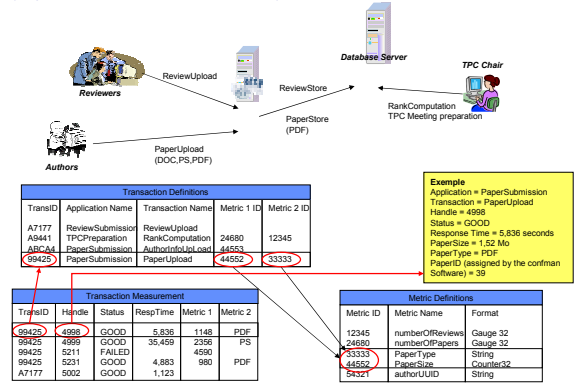
- Le problème
 - Toutes (ou presque) les applications ont besoin de données de performance et de disponibilité :
 - Souvent du bricolage dans les applications,
 - Sémantique des transactions sont spécifiques aux applications
 - Ex. bankTransfer(), queryAccount(), addToShoppingCart(), googleSearch(), compile(), generateNewSoftwareVersion(), collectAllUserVotes()
- Solution
 - Modèle Universel Unique pour la mesure de performance de transactions & de disponibilité
 - Objets abstraits pour
 - UnitOfWork
 - D'un appel de méthode simple à
 - Une transaction longue multi-niveaux imbriquée multi-sites
 - APIs pour développeurs d'applications
 - Utilisation du modèle
 - Instrument leurs application
 - Implémentations d'agent par fournisseurs de « framework »
 - Classes ARM concrètes et traitements associés
 - gère les données de performance
 - Ex. journalise, établit de graphes et les publie sur le web, nourrit la gestion de fautes, ...



Un exemple ArmTransaction usage Model



Exemple du monitoring d'un système de conférence (inspiré du manuel de référence d'ARM)



ARM : synthèse

- Petit mais costaud !
 - Une spec normative qui tiend en 100 pages (dont du javadoc) et qui est compréhensible par un être binaire à BAC+10... le bonheur
 - Hyper utile
- Exploitée ????
 - Peu d'implémentations disponibles sur Internet
 - Guère que celle de l'Open Group
 - ++ Celle de MADYNES

Gestion à base d'XML

- Définition
 - Utilisation d'XML et des technologies associées comme support à la supervision
- Très à la mode aujourd'hui
 - Même l'IRTF & l'IETF s'y intéressent
- De véritables approches
 - WBEM en est la première
 - Configuration Management en profite vraiment
 - Du micro-device : T68i avec SyncML
 - Au macro-device : Routeurs JUniper avec Junoscript
 - Et bien sûr dans les plateformes
 - Ex: OSS-J

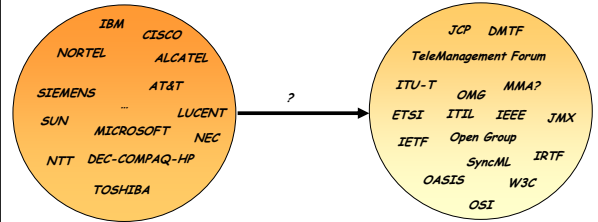
```
GET /context[@name="router.nowhere.com"
and @port="161"
and @community="public"]
/IF MB/IfEntry/IfOperStatus="up"
and ((IfInErrors * 10000 > IfInOctets)
or (IfOutErrors * 10000 > IfOutOctets))/IfDescr
```

Super intéressant, mais un cours à elle toute seule !

Plan

- Supervision des réseaux & services
 - Quelques rappels
- Évolution des espèces
 - Du GDMOsaure au DISMANopithèque
 - Leçons !
- Les solutions actuelles & à venir
 - JMX,
 - WBEM & CIM
 - ARM
 - XML Management
- Synthèse
- Futur
- Q&R.

Synthèse



- Combien de consortiums de normalisation peut-on créer à partir de n entreprises sachant qu'une norme requiert au moins 2 entreprises ?
- (2) Enumérer les avantages

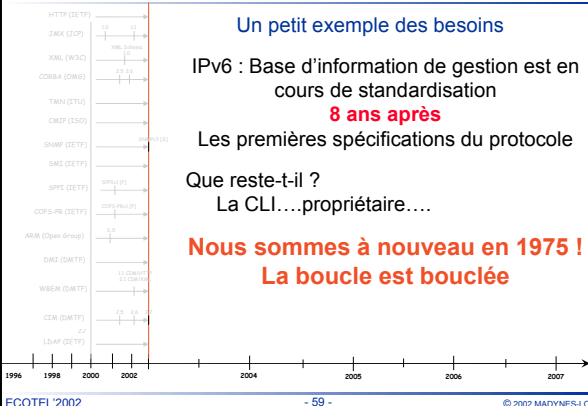
Synthèse

- « Le progrès, c'était mieux avant ! »
 - On a pas fait mieux depuis GDMO/CMIS
 - On a pas fait pire non plus ☺
- La supervision profite largement des évolutions des intergiciels/modèles et langages de programmation
 - JMX en est l'exemple le plus flagrant
 - Entre le OUAIB et le Java, qui s'en va ?
- Au vu de JMX et de WBEM
 - La normalisation OSI a été merveilleuse
- Des intergiciels et modèles de qualité existent pour couvrir les besoins à court terme
 - Toolkits d'instrumentation
 - Activités foisonnante sur de nombreuses choses
 - Policy, XML, Java, P2P, ...

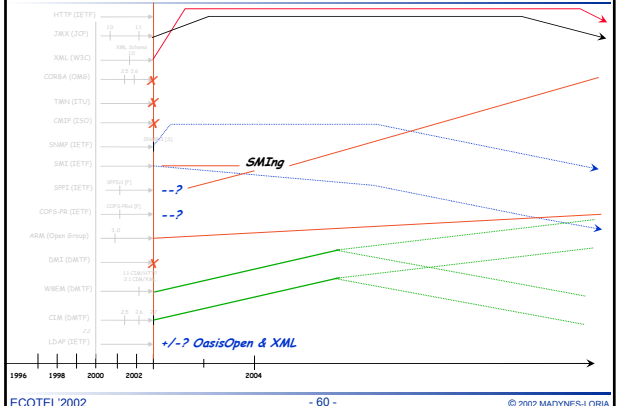
Plan

- Supervision des réseaux & services
 - Quelques rappels
- Évolution des espèces
 - Du GDMOsaure au DISMANopithèque
 - Leçons !
- Les solutions actuelles & à venir
 - JMX,
 - WBEM & CIM
 - ARM
 - XML Management
- Synthèse
- Futur
- Q&R.

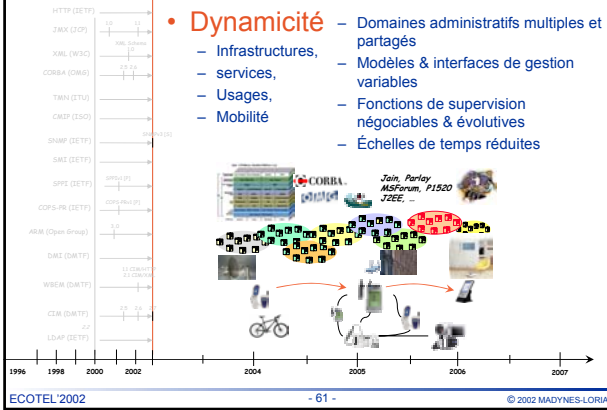
(No) Future



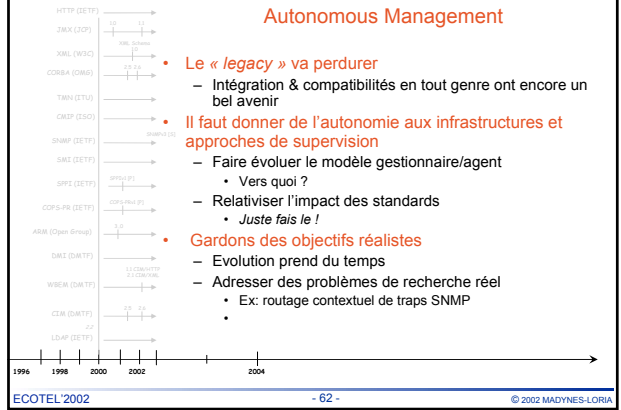
Et demain ? Les normes en tendance



La vision du projet MADYNES



La vision du projet MADYNES



R f rences

- [Sch nw lder02] J. Sch nw lder, *Evolution of Open Source SNMP Tools*, in *Proc. SANE 2002 Conference*, May 2002
- [Lewis01] L. Lewis, *Service Level Management for Enterprise Networks*, 2001.